GRVI Phalanx: A Massively Parallel RISC-V® FPGA Accelerator Framework

⇨ A 1680-core, 26 MB SRAM Parallel Processor Overlay on Xilinx UltraScale+ VU9P

Jan Gray | Gray Research LLC | Bellevue, WA | jan@fpga.org | http://fpga.org

Datacenter FPGA accelerators are mainstream

- MS Catapult, Amazon AWS F1, Intel += Altera, Baidu
- Massively parallel, specialized, connected, versatile
- High throughput, low latency, energy efficient

But two hard problems

- Software: Porting & *maintaining* workload as accelerator
- Hardware: Compose 100s of cores, 25-100G NICs, many DRAM/HBM channels, with easy timing closure

GRVI Phalanx: FPGA accelerator framework

- **GRVI**: FPGA-efficient RISC-V processing element
- **Phalanx:** CPU/accelerator/IO fabric: **clusters** of PEs, SRAMs, accelerators, DRAM/IO controllers on a …
- **Hoplite NoC**: FPGA-optimal directional 2D torus NoC
- Local shared memory, global message passing

Software-first, software-mostly accelerators

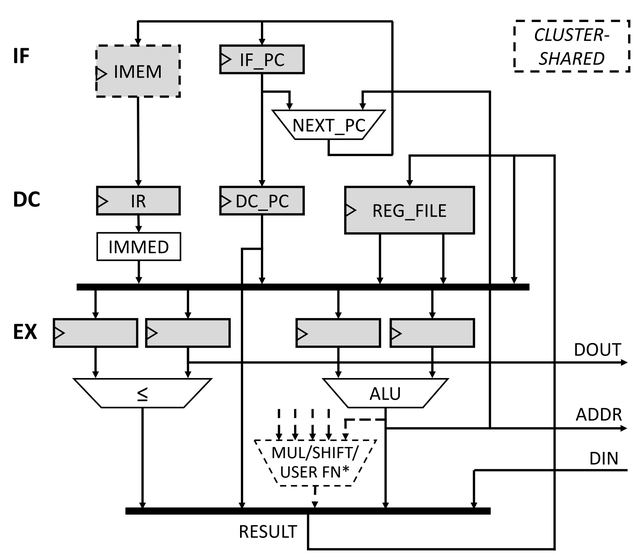
- Run your parallel software on 100s of soft processors
- Add custom function units/cores/memories to suit
- More 10 sec recompiles, fewer 10 hr synth/place/route
- Complements high level synthesis & OpenCL→FPGA flows

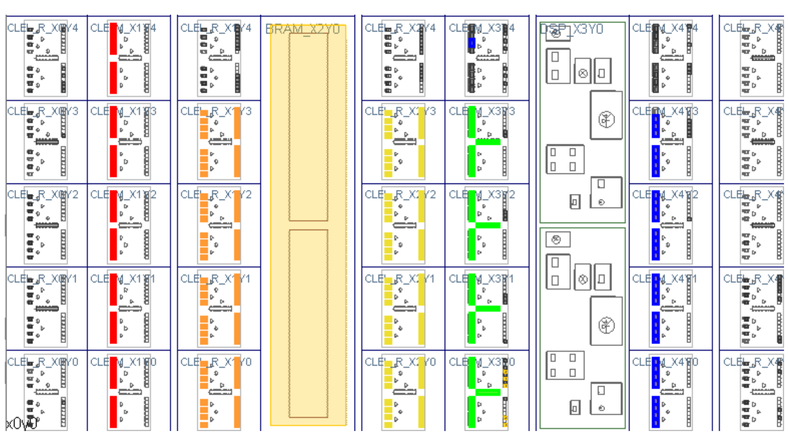
FPGA soft processor area and energy efficiency

- Simpler CPUs → more CPUs → more memory parallelism
- Jan's Razor: "In a chip-multiprocessor, cut *inessential* resources from each CPU, to maximize CPUs per die."
- Share function units in the cluster
- Sweat every LUT

GRVI: *austere* RISC-V processing element (PE)

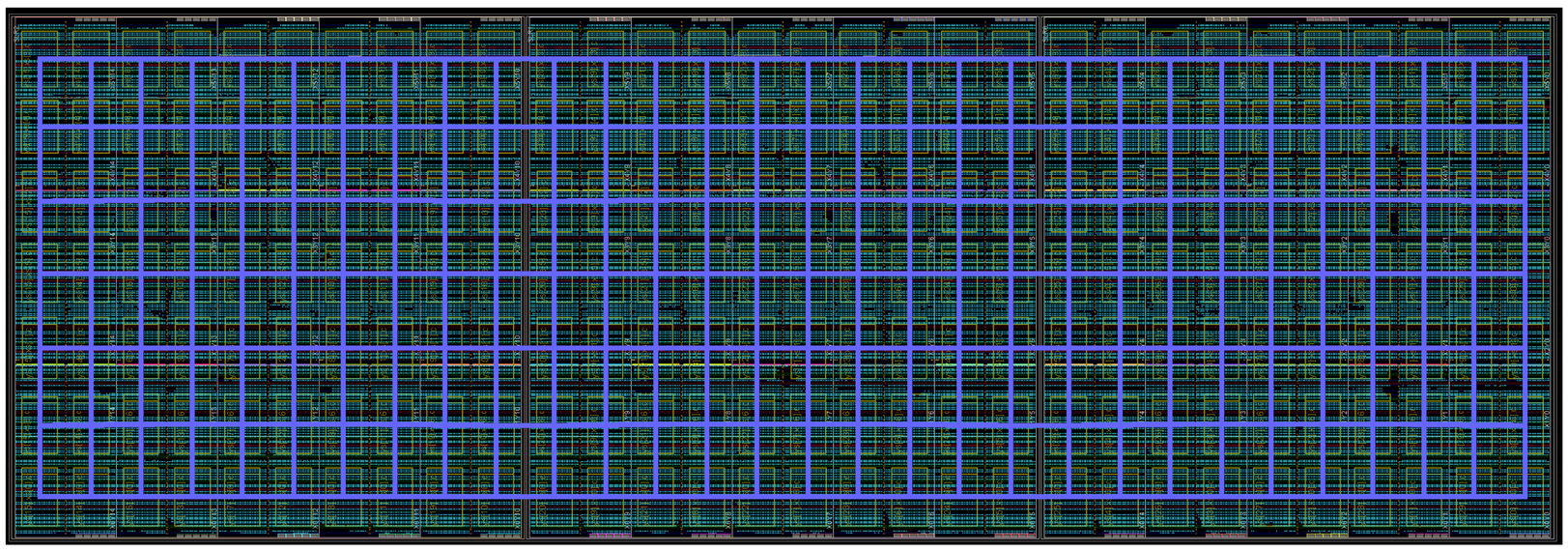
- *User mode* RV32I, *minus* all CSRs *plus* -M `mul*` (cluster DSP), -A `lr/sc` (cluster RAM banks)
- 3 pipeline stages (fetch, decode, execute)
- 2 cycle loads; 3 cycle taken branches/jumps
- Painstakingly technology mapped and floorplanned
- **Typically 320 LUTs** @ 375 MHz ≈ **0.7 MIPS/LUT**





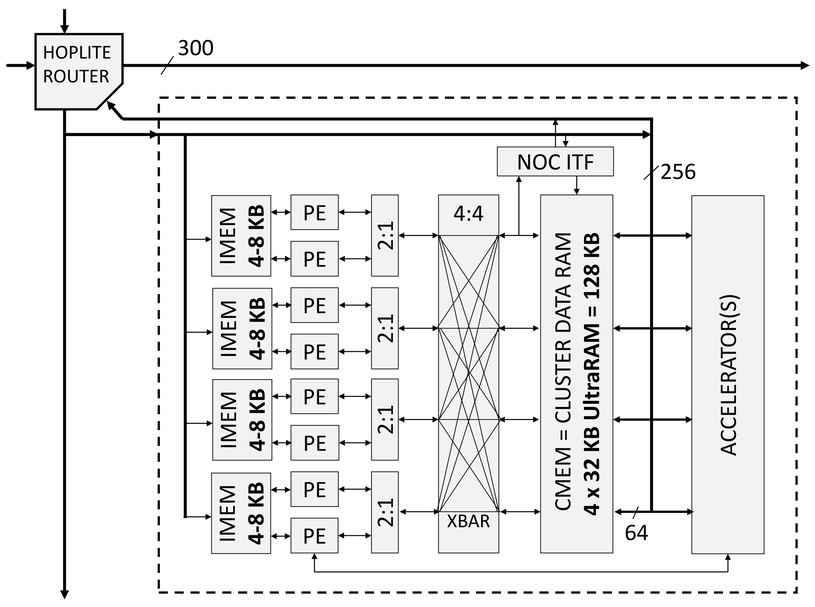
Phalanx: many *clusters* of PEs/accelerator/IOs

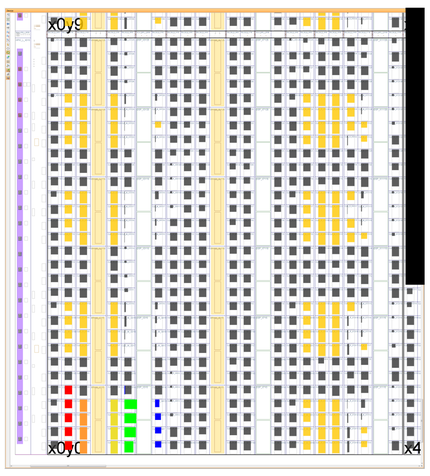
- Make it easy to exploit massive FPGA BRAM bandwidth
- Configurable, heterogeneous mixes of clusters
- Interconnected by an extreme bandwidth **Hoplite NoC**
- PGAS: partitioned global address space
- **32 byte/cycle/cluster** message passing between clusters, standalone accelerators, IO and DRAM controllers
- So far, no caches – small kernel instruction RAMs and multiported shared cluster RAMs
- Planned: "last level" caches at DRAM controller bridges



Cluster: 0-8 PEs, 32-128 KB RAM, accel, router

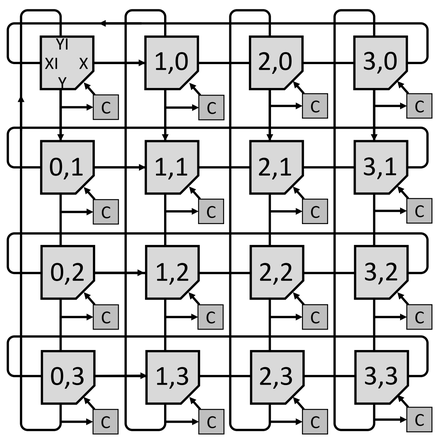
- Compose cores & accelerator(s), & send/receive 32 byte messages via multiported banked cluster shared RAM
- Typ. 4000 LUTs + 8-12 BRAM + 0-4 URAM + 4-8 DSP

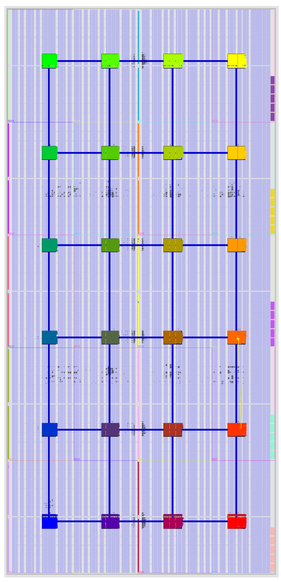




Hoplite: FPGA-optimal 2D torus router and NoC

- Rethink FPGA NoC router architecture
- No segmentation/flits, no VCs, no buffering, no credits
- (Default) deflecting dimension order routing
- Simple 3×2 switch → frugal → ultra wide → high BW
- Configurable routing, link pipelining, multicast





- *A 400 MHz 4×6×256b Hoplite NoC, 100 Gb/s links, uses 2.7% of KU040*

Hoplite router: Xilinx, Intel *optimal* area×delay

- One LUT/bit/router; one FF-*wire*-LUT-FF delay/router = **1%** of area × delay of prior FPGA-optimized VC routers
- Featherweight router-client interface – zero LUTs/bit
- 8b-1024b wide → **4-400 Gb/s links** → Tb/s bisec BW
- Perfect for Intel HyperFlex pipelined interconnect
- *Everything is interconnected / IP site doesn't matter much*

AWS F1 uses Xilinx Virtex UltraScale+ VU9P

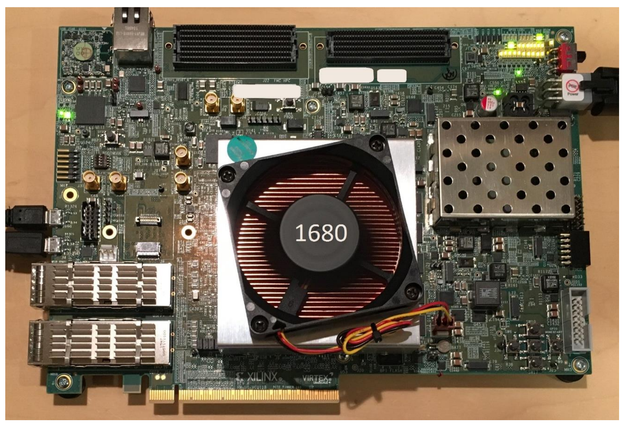
- 1.2M LUTs, 2160 4KB BRAMs, 960 32KB URAMs, 7K DSPs

A 1680-core GRVI Phalanx on VU9P (above)

- **30×7 clusters** of { **8 GRVI PEs**, 128 KB CRAM, router }
- 1680 cores, 26 MB cluster RAM, 210 300b routers

Running in Xilinx VCU118

- 250 MHz; 420 GIPS; 2.5 TB/s SRAM; 0.9 Tb/s NoC bis.BW
- 31-40 W → 18-24 mW/processor
- 67% of LUTs; 88% of URAM; 39% of BRAM; 12% of DSP
- 1300 BRAMs + 6000 DSPs available for accelerators
- First kilocore RISC-V SoC, most ever 32b RISC PEs/chip



Accelerated parallel programming models

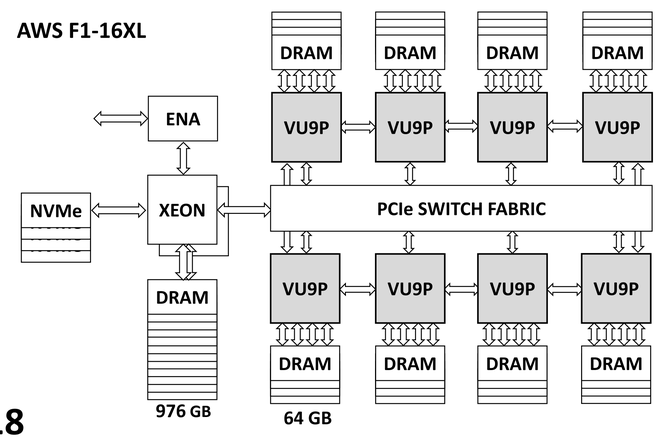
- SPMD/MIMD code w/ small kernels, local or PGAS shared memory, message passing, memcpy/RDMA DRAM
- Current: multithread C++ with message passing runtime, built with GCC for RISC-V RV32IMA
- Future: OpenCL, 'Gatling gun' packet processing / P4, message passing / streaming data / KPNs
- Accelerated with plug-in-custom FUs, RAMs, AXI cores

Recent work

- AXI4 & AXI-Stream bridges, Xilinx IP Integrator support
- Zynq PS and DRAM controllers interfacing
- 80-core edu edition for Xilinx Z7020 / PYNQ-Z1 ($65)
- Hoplite NoC auto-segmentation → greater bandwidth

Work in progress

- **Target AWS F1.2XL & F1.16XL**: add PCIe DMA bridge, 4 DRAM/RDMA/LLC$ chans, inter-FPGA message passing
- Up to 10,000-GRVI Phalanx per F1.16XL instance
- 64-bit GRVI64 to directly address 1.5 TB F1.16XL
- PYNQ-Z1 and F1 kits and general availability



2018

- Arria 10 SP FPU-DSPs, Stratix 10 Hyperflex-Hoplite NoC
- OpenCL, HBM2 memory systems, 25-100 GbE NICs